1. v, u are arbitrarily selected
   - we don't know if there is some w, z with $> x$ or $< x$ edge-disjoint paths
   - However, we do know that at most, G must be x-edge connected
   - Since vertex connectivity is bounded above by edge connectivity, G is also at most x-connected

$$\boxed{\kappa'(G) \leq x}$$

$$1 \leq \boxed{\kappa(G) \leq \kappa'(G) \leq x}$$

as G is defined to be connected

(2) - We know that the number of edge-disjoint paths bounds our edge-connectivity

- We seek some minimum set of edge-disjoint paths for some u, v to get the edge-connectivity of G

Pseudocode:

```
GetConnectivity(Graph G)
 minSize = ∞
 for all u ∈ V(G)
    for all v ∈ V(G), v ≠ u
        paths = getAllPaths(G, u, v)
        if paths.size() < minSize
           minSize = paths.size()
 return minSize
```

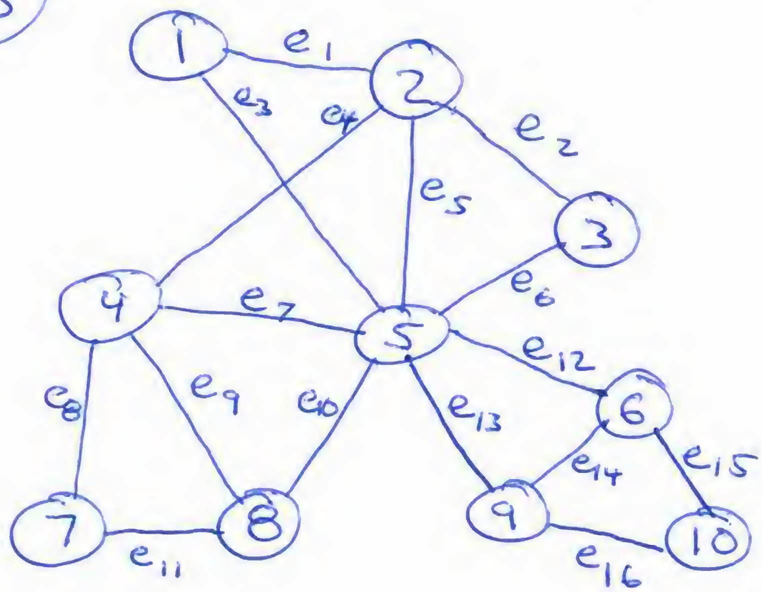
---

complexity $= \underbrace{|V||V|}_{\text{nested loops}} \underbrace{(|V| + |E|)}_{\text{assumption for getAllPaths()}}$

$\approx O(n^3)$

polynomial time

3



- We observe that vertex 5 is a cut vertex, so no open-ear decomposition exists
- The graph is connected however, so we have $\boxed{K(G) = 1}$

Closed-ear decomposition:

$P_0 = \{e_1 e_2 e_6 e_3\}$

$P_1 = \{e_{10} e_{11} e_8 e_4\}$

$P_2 = \{e_5\}$

$P_3 = \{e_7\}$

$P_4 = \{e_9\}$

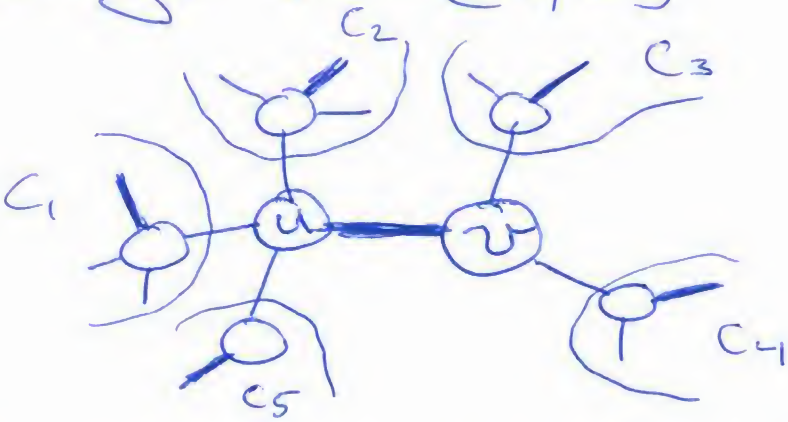
$P_5 = \{e_{12} e_{15} e_{16} e_{13}\}$

$P_6 = \{e_{14}\}$

- All ears open except for P_5
- G has a closed-ear decomposition, so G is 2-edge-connected

$\boxed{K'(G) = 2}$

(4) - Consider graph G with perfect match M, and some matched edge $m=(u,v)$



- Removing vertices (u,v) will create $d(u)+d(v)-2$ components, each with a perfect match and therefore even number of vertices
- If our match M is not unique, then for some such (u,v) there exists a match without (u,v) but with $m_1=(u,x)$ and $m_2=(v,y)$

- Consider C_1+u, a component with a perfect match in $G-(u,v)$ where $x \in V(C_1)$
- As edge (u,x) now saturates x, we now have a subgraph of G with an odd number of vertices and can therefore not be perfectly matched, a contradiction □

(5) $\forall v \in V(G): d(v)$ is even iff
for all B_i, $\forall u \in V(B_i): d(u)$ is even,
where B_i are maximal biconnected components

if $\forall B_i, \forall u \in V(B_i): d(u)$ even
$\Rightarrow \forall v \in V(G): d(v)$ is even

- Obviously, for a $u \in B_i$ which are not articulationpoints, the degrees in a BiCC and G will both be even
- For articulation points, their degree in G is the sum of degrees in each BiCC, and a sum of even numbers will necessarily also be even

if $\forall v \in V(G): d(v)$ is even $\Rightarrow$
$\forall B_i, \forall u \in V(B_i): d(u)$ is even

We'll do induction on the number of BiCCs in G

Base: one BiCC $\rightarrow$ obviously as $B_1 = G$
then $\forall u \in V(B_1): d(u)$
$= \forall v \in V(G): d(v)$

5 cont.

Hypothesis: For some $P(k) = H$ with k BiCCs and even degrees in H, all BiCCs taken as subgraphs have even degrees

Step: We consider some $P(n) = G$ with $n > k$ BiCCs

- We create H by removing some B_i with at most one articulation point
- We invoke I.H. on H
- We consider adding B_i back into G to show our hypothesis holds

Case 1: B_i was disconnected from the rest of G, its removal didn't affect any degrees in H and its degrees in G and internal to itself are equal $\rightarrow$ even

Case 2: B_i was connected to G through some articulation point a_i. By the degree sum formula it must have an even number of edges in B_i and also the rest of G. We invoke our I.H. on $G - B_i$. We then note that adding back B_i does not ~~a~~affect degrees in any other vertex except a_i, which we already know must be